Unsupervised Classification with Dependency Based Word Spaces

Klaus Rothenhäusler and Hinrich Schütze
Institute for Natural Language Processing
University of Stuttgart
Stuttgart, Germany
{Klaus.Rothenhaeusler, Hinrich.Schuetze}@ims.uni-stuttgart.de

Abstract

We present the results of clustering experiments with a number of different evaluation sets using dependency based word spaces. Contrary to previous results we found a clear advantage using a parsed corpus over word spaces constructed with the help of simple patterns. We achieve considerable gains in performance over these spaces ranging between 9 and 13% in absolute terms of cluster purity.

1 Introduction

Word space models have become a mainstay in the automatic acquisition of lexical semantic knowledge. The computation of semantic relatedness of two words in such models is based on their distributional similarity. The most crucial way in which such models differ is the definition of distributional similarity: In a regular word space model the observed distribution concerns the immediate neighbours of a word within a predefined window to the left and right (Schütze, 1992; Sahlgren, 2006). Early on in the development as an alternative models were proposed that relied on the similarity of the distribution of syntactic relations (Hindle, 1990; Padó and Lapata, 2007). More recently the distribution of the occurrence within simple patterns defined in the form of regular expressions that are supposed to capture explicit semantic relations was explored as the basis of distributional similarity (Almuhareb and Poesio, 2004).

Whereas dependency based semantic spaces have been shown to surpass other word space models for a number of problems (Padó and Lapata, 2007; Lin, 1998), for the task of categorisation simple pattern based spaces have been shown to perform equally good if not better (Poesio and Almuhareb, 2005b; Almuhareb and Poesio, 2005b). We want to show that dependency based spaces also fare better in these tasks if the dependency relations used are selected reasonably. At the same time we want to show that such a system can be built with freely available components and without the need to rely on the index of a proprietary search engine vendor.

We propose to use the web acquired data of the ukWaC (Ferraresi et al., 2008), which is huge but still manageable and comes in a pre-cleaned version with HTML markup removed. It can easily be fed into a parser like MiniPar which allows for the subsequent extraction of dependency relations of different types and complexity. In particular we work with dependency paths that can reach beyond direct dependencies as opposed to Lin (1998) but in the line of Pado and Lapata (2007). In contrast to the latter, however, different paths that end in the same word are not generally mapped to the same dimension in our model. A path in a dependency graph can pass through several nodes and encompass different relations.

We experimented with two sets of nouns previously used in the literature for word clustering. The nouns in both sets are taken from a number of different WordNet categories. Hence, the task consists in clustering together the words from the same category. By keeping the clustering algorithm constant, differences in performance can be attributed to the differences of the word representations.

The next section provides a formal description of our word space model. Section 3 reports on our clustering experiments with two sets of concepts used previously to evaluate the categorisation abilities of word spaces. Section 4 discusses these re-

Proceedings of the EACL 2009 Workshop on GEMS: GEometical Models of Natural Language Semantics, pages 17–24,
Athens, Greece, 31 March 2009. ©2009 Association for Computational Linguistics

sults and draws some conclusions.

2 Word Space Construction

We follow the formalisation and terminology developed in Pado and Lapata (2007) according to which a dependency based space is determined by the sets of its basis elements B and targets T that form a matrix $M = B \times T$, a similarity function S that assigns a real-valued similarity measure to pairs of elements from T, the association measure A that captures the strength of the relation between a target and a basis element, the context selection function *cont*, the basis mapping function μ and the path value function v. Our set of targets is always a subset of the lemmas output by MiniPar. The remaining elements are defined in this section. We use π to denote a path in a dependency graph which is conceived of as an undirected graph for this purpose. So, in general a dependency path has an upward and downward part where one can have length zero. All the paths used to define the contexts for target words are anchored there, i.e. they start from the target.

In choosing the context definitions that determine what dependency paths are used in the construction of the word vectors, we oriented ourselves at the sets proposed in Pado and Lapata (2007). As Pado and Lapata (2007) achieved their best results with it we started from their medium sized set of context definitions, from which we extracted the appropriate ones for our experiments and added some that seemed to make sense for our purposes: As our evaluation sets consist entirely of nouns, we used only context definitions that start at a noun. Thereby we can ensure that only nominal uses are recorded in a word vector if a target word can have different parts of speech. The complete set of dependency relations our context selection function *cont* comprises is given in Figure 1 along with an example for each.

We only chose paths that end in an open word class assuming that they are more informative about the meaning of a target word. Paths ending in a preposition for instance, as used by Pado and Lapata (2007), were not considered. For the same reason we implemented a simple stop word filter that discards paths ending in a pronoun, which are assigned the tag `N` by MiniPar just like any other noun.

On the other hand we added the relation between a prepositional complement and the noun it modifies (appearing as relation IX in Figure 1) as a close approximation of the pattern used by (Almuhareb and Poesio, 2004) to identify attributes of a concept as detailed in the next section. Path specifications X and XI are also additions we made that are thought to gather additional attribute values to the ones already covered by III.

As a basis mapping function μ we used a generalisation of the one used by Grefenstette (1994) and Lin (1998). They map a dependency between two words to a pair consisting of the relation label l and the end word of the dependency $end(\pi)$. As we use paths that span more than a single relation, this approach is not directly applicable to our setup. Instead we use a mapping function that maps a path to the sequence of edge labels through which it passes combined with the end word:

$$\mu(\pi) = (l(\pi), end(\pi))$$

where $l(\cdot)$ is a labelling function that returns the sequence of edge labels for a given path. With this basis mapping function the nodes or words respectively through which a path passes are all neglected except for the node where the path ends. So, for the noun *human* the sequence *human and mouse genome* as well as the sequence *human and chimpanzee genome* increase the count for the same basis element `:N:conj:N:*:N:nn:N:genome`. Here we use a path notation of the general form:

$$(:\texttt{POS}:\texttt{rel}:\texttt{POS}:\{\texttt{word}, *\})^n$$

where `POS` is a part of speech, `rel` a relation and `word` a node label, i.e. a lemma, all as produced by MiniPar. The length of a path is determined by n and the asterisk (`*`) indicates that a node label is ignored by the basis mapping function.

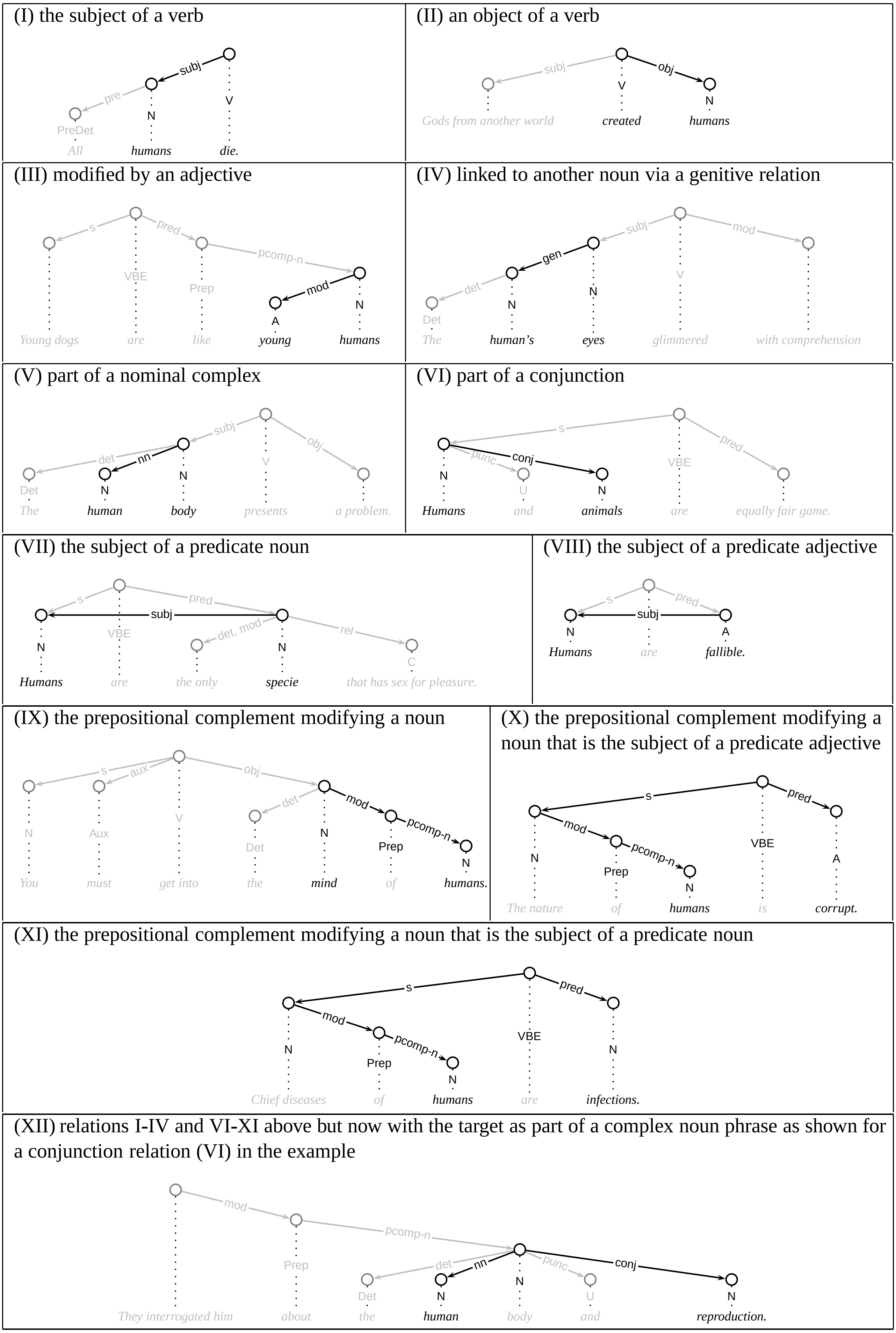
As an alternative we experimented with a lexical basis mapping function that maps a path to its end word:

$$\mu(\pi) = end(\pi)$$

This reduces the number of dimensions considerably and yields semantic spaces that are similar to window based word spaces. As this mapping function consistently delivered worse results, we dropped it from our evaluation.

Considering that (Padó and Lapata, 2007) only reported very small differences for different path valuation functions, we only used a constant valuation of paths:

$$v_{\text{const}}(\pi) = 1$$



(I) the subject of a verb

(II) an object of a verb

(III) modified by an adjective

(IV) linked to another noun via a genitive relation

(V) part of a nominal complex

(VI) part of a conjunction

(VII) the subject of a predicate noun

(VIII) the subject of a predicate adjective

(IX) the prepositional complement modifying a noun

(X) the prepositional complement modifying a noun that is the subject of a predicate adjective

(XI) the prepositional complement modifying a noun that is the subject of a predicate noun

(XII) relations I-IV and VI-XI above but now with the target as part of a complex noun phrase as shown for a conjunction relation (VI) in the example

Figure 1: Context definitions used in the construction of our word spaces. All examples show contexts for the target *human*. Greyed out parts are just for illustrative purposes and have no impact on the word vectors. The examples are slightly simplified versions of sentences found in ukWaC.

Thus, an occurrence of any path, irrespective of length or grammatical relations that are involved, increases the count of the respective basis element by one.

We implemented three different association functions, A, to transform the raw frequency counts and weight the influence of the different co-occurrences. We worked with an implementation of the *log likelihood ratio* (g-Score) as proposed by Dunning (1993) and two variants of the *t-score*, one considering all values (t-score) and one where only positive values (t-score$^+$) are kept following the results of Curran and Moens (2002). We also experimented with different frequency cutoffs removing dimensions that occur very frequently or very rarely.

3 Evaluation

For all our experiments we used the ukWaC corpus[1] to construct the word spaces, which was parsed using MiniPar. The latter provides lemma information, which we used as possible target and context words. The word vectors we built from this data were represented as pseudo documents in an inverted index. To our knowledge the experiments described in this paper are the first to work with a completely parsed version of the ukWaC.

For the evaluation the word vectors for the test sets were clustered into a predefined number of clusters corresponding to the number of concept classes from which the words were drawn. All experiments were conducted with the CLUTO toolkit (Karypis, 2003) using the repeated bisections clustering algorithm with global optimisation and the cosine as a distance measure to maintain comparability with related work, e.g. Baroni et al. (2008).

As the main evaluation measure we used purity for the whole set as supplied by CLUTO. For a clustering solution Ω of n clusters and a set of classes C, purity can be defined as:

$$\text{purity}(\Omega, C) = \frac{1}{n}\sum_k \max_j |\omega_k \cap c_j|$$

where ω_k denotes the set of terms in a cluster and c_j the set of terms in a class. This aggregate measure of purity corresponds to the weighted sum of purities for the individual clusters, which is defined as the ratio of items in a cluster that belong to the majority class. The results for the two test sets we used are described in the following two subsections.

3.1 Results for 214 nouns from Almuhareb and Poesio (2004)

The first set we worked with was introduced by Almuhareb and Poesio (2004) and consists of 214 nouns from 13 different categories in WordNet. In the original paper the best results were achieved with vector representations built from concept attributes and their values as identified by simple patterns. For the identification of attribute values of a concept C the following pattern was used

“[a|an|the] * C [is|was]”

It will find instances such as *an adult human is* identifying *adult* as a value for an attribute (age) of [HUMAN] (we use small capitals enclosed in square brackets to denote a concept). Attributes themselves are searched with the pattern

“the * of the C [is|was]”

A match for the concept [HUMAN] would be *the dignity of the human is*, which yields *dignity* as an attribute. These patterns were translated into queries and submitted to the Google[2] search engine.

We compare our dependency based spaces with the results achieved with the pattern based approach in Table 1.

association measure	g-score	t-score	t-score$^+$
dependency based space	77.1%	85.5%	96.7%
window based space	84.1%	82.7%	89.3%
pattern based space	-	-	85.5%

Table 1: Categorisation results for the 214 concepts and 13 classes proposed in Almuhareb and Poesio (2004), which is also the source of the result for the pattern based space. They only used t-score$^+$. The numbers given are the best accuracies achieved under the different settings.

For the window based space we used the best performing in a free association task with a window size of six words to each side and all the

[1] http://wacky.sslmit.unibo.it

[2] http://www.google.com

context	accuracy	# dimensions
(I)	82.2%	7359
(II)	92.5%	6680
(III)	88.3%	45322
(IV)	–	37231
(V)	82.2%	240157
(VI)	95.3%	93917
(VII)	86.9%	45527
(VIII)	77.1%	5245
(IX)	91.6%	87765
(X)	–	2186
(XI)	–	6967
(XII)	93.0%	188763

Table 2: Clustering results using only one kind of path specification. For (IV), (X) and (XI) purity values are missing because vectors for some of the words could not be built.

words that appeared at least two times as dimensions ignoring stop words. The effective dimensionality of the so built word vectors is 417 837.

The results for the dependency based spaces were built by selecting all paths without any frequency thresholds which resulted in a set of 767 119 dimensions.

As can be seen, both window and dependency based spaces exceed the pattern based space for certain association measures. But the dependency space also has a clear advantage over the window based space. In particular the t-score$^+$ measure yields very good results. In contrast the g-score offers the worst results with the t-score retaining negative values somewhere in between. For our further experiments we hence used the t-score$^+$ association measure.

3.1.1 Further Analysis

We ran a number of experiments to quantify the impact the different kinds of paths have on the clustering result. We first built spaces using only a single kind of path to find out how good each performs on its own. The result can be found in Table 2. For some of the words in the evaluation set no contexts could be found when only one of the two most complex context specifications (X), (XI) was used or when the context was reduced to the genitive relation (IV). Apart from that the results suggest that even a single type of relation on its own can prove highly effective. Especially the conjunctive relation (VI) performs very well with a purity value of 95.3%.

removed context	accuracy
(I)	97.2%
(II)	97.7%
(III)	97.2%
(IV)	97.2%
(V)	98.1%
(VI)	96.3%
(VII)	97.2%
(VIII)	97.2%
(IX)	96.7%
(X)	97.2%
(XI)	97.2%
(XII)	96.7%

Table 3: Clustering results for spaces with one context specification removed.

To further clarify the role of the different kinds of contexts, we ran the experiment with word spaces where we removed each one of the twelve context specifications in turn. The results as given in Table 3 are a bit astonishing at first sight: Only the removal of the conjunctive relation actually leads to a decrease in performance. All the other contexts seem to be either redundant – with performance staying the same when they are removed – or even harmful – with performance increasing once they are removed. Having observed this, we tried to remove further context specifications and surprisingly found that the best performance of 98.1% can be reached by only including the conjunction (VI) and the object (II) relations. The dimensionality of these vectors is only a fraction of the original ones with 100 597.

The result for the best performing dependency based space listed in the table is almost perfect. Having a closer look at the results reveals that in fact only four words are put into a wrong cluster. These words are: *lounge, pain, mouse, oyster*.

The first is classified as [BUILDING] instead of [FURNITURE]. In the case of *lounge* the misclassification seems to be attributable to the ambiguity of the word which can either denote a piece of furniture or a waiting room. The latter is apparently the more prominent sense in the data. In this usage the word often appears in conjunctions with *room* or *hotel* just like *restaurant, inn* or *clubhouse*.

Pain is misclassified as an [ILLNESS] instead of a [FEELING] which is at least a close miss. The misclassification of *mouse* as a [BODY PART] seems rather odd on the other hand. The reason for it becomes apparent when looking at the most descriptive and discriminating features of the [BODY PART] cluster: In both lists the highest in the ranking is the dimension `:N:mod:A:left`, i.e. *left* as an adjectival modifier of the word in question. The prominence of this particular modification is of course due to the fact that a lot of body parts come in pairs and that the members of these pairs are commonly identified by assigning them to the left or right half of the body. Certainly, the word *mouse* enters this cluster not through its sense of *mouse*^{1} as an animal but rather through its sense of *mouse*^{2} as a piece of computer equipment that has two buttons, which are also referred to as the left and right one. Unfortunately, MiniPar frequently resolves *left* in a wrong way as a modifier of *mouse* instead of *button*.

Finally for *oyster* which is put into the [EDIBLE FRUIT] instead of the [ANIMAL] cluster it is conspicuous that *oyster* is the only sea animal in the evaluation set and consequently it rarely occurs in conjunctions with the other animals. Conjunctions, however, seem to be the most important features for defining all the clusters. Additionally *oyster* scores low on a lot of dimensions that are typical for a big number of the members of the animal cluster, e.g. `:N:obj:V:kill`.

3.2 Results for 402 words from Almuhareb and Poesio (2005a)

In Poesio and Almuhareb (2005a) a larger evaluation set is introduced that comprises 402 nouns sampled from the hierarchies under the 21 unique beginners in WordNet. The words were also chosen so that candidates from different frequency bands and different levels of ambiguity were represented. Further results using this set are reported in Almuhareb and Poesio (2005b). The best result was obtained with the attribute pattern alone and filtering to include only nouns. We tried to assemble word vectors with the same patterns based on the ukWaC corpus. But even if we included both patterns, we were only able to construct vectors for 363 of the 402 words. For 118 of them the number of occurrences, on which they were based, was less than ten. This gives an impression of the size of the index that is necessary for such an approach. To date such an immense amount of data is only available through proprietary search engine providers. This makes a system dependant upon the availability of an API of such a vendor. In fact the version of the Google API on which the original experiments relied has since been axed. Our approach circumvents such problems.

We ran analogous experiments to the ones described in the previous section on this evaluation set, now producing 21 clusters. The results given in Table 4 are for a dependency space without any frequency thresholds and the complete set of context specifications as defined above. The settings for the window based space were also the same (6 words to each side). Again the results achieved with the t-score$^+$ association were clearly superior to the others and were used in all the following experiments. Unsurprisingly, for this more difficult task the performance is not as good as for the smaller set but nevertheless the superiority of the dependency based space is clearly visible with an absolute increase in cluster purity of 8.2% compared with the pattern based space.

association measure	g-score	t-score	t-score$^+$
dependency based space	67.9%	67.2%	79.1%
window based space	65.7%	60.7%	67.9%
pattern based space	-	-	70.9%

Table 4: Categorisation results for the 402 concepts and 21 classes proposed in Almuhareb and Poesio (2005a) which is also the source of the result for the pattern based space. The numbers given are the best accuracies achieved under the different settings.

3.2.1 Further Analysis

Again we ran further experiments to determine the impact of the different kinds of relations. The removal of any single context specification leads to a performance drop with this evaluation set. The smallest decrease is observed when removing context specification XII. However, as we had seen in the previous experiment with the smaller set that only two context specifications suffice to reach peak performance, we conducted another experiment where we started from the best performing space constructed from a single context specification (the conjunction relation, VI) and successively added the specification that led to the biggest performance gain. The crucial results are

majority class	concepts
solid	tetrahedron, salient, ring, ovoid, octahedron, knob, icosahedron, fluting, dome, dodecahedron, cylinder, cuboid, cube, crinkle, concavity, *samba, coco, nonce, divan, ball, stitch, floater, trove, hoard, mouse*
time	yesteryear, yesterday, tonight, tomorrow, today, quaternary, period, moment, hereafter, gestation, future, epoch, day, date, aeon, *stretch, snap, throb, straddle, nap*
motivation	wanderlust, urge, superego, obsession, morality, mania, life, impulse, ethics, dynamic, conscience, compulsion, *plasticity, opinion, acceptance, sensitivity, desire, interest*
assets	wager, taxation, quota, profit, payoff, mortgage, investment, income, gain, fund, credit, capital, allotment, allocation, *possession, inducement, incentive, disincentive, deterrence, share, sequestrian, cheque, check, bond, tailor*
district	village, town, sultanate, suburb, state, shire, seafront, riverside, prefecture, parish, metropolis, land, kingdom, county, country, city, canton, borough, borderland, anchorage, *tribe, nation, house, fen, cordoba, faro*
legal document	treaty, statute, rescript, obligation, licence, law, draft, decree, convention, constitution, bill, assignment, *commencement, extension, incitement, caliphate, clemency, venture, dispensation*
physical property	weight, visibility, temperature, radius, poundage, momentum, mass, length, diameter, deflection, *taper, indentation, droop, corner, concavity*
social unit	troop , team, platoon, office, legion, league, household, family, department, confederacy, company, committee, club, bureau, brigade, branch, agency
atmospheric phenomenon	wind, typhoon, tornado, thunderstorm, snowfall, shower, sandstorm, rainstorm, lightning, hurricane, fog, drizzle, cyclone, crosswind, cloudburst, cloud, blast, aurora, airstream, *glow*
social occasion	wedding, rededication, prom, pageantry, inaugural, graduation, funeral, fundraiser, fiesta, fete, feast, enthronement, dance, coronation, commemoration, ceremony, celebration, *occasion, raffle, beano*
monetary unit	zloty, yuan, shilling, rupee, rouble, pound, peso, penny, lira, guilder, franc, escudo, drachma, dollar, dirham, dinar, cent
tree	sycamore, sapling, rowan, pine, palm, oak, mangrove, jacaranda, hornbeam, conifer, cinchona, casuarina, acacia, *riel*
chemical element	zinc, titanium, silver, potassium, platinum, oxygen, nitrogen, neon, magnesium, lithium, iron, hydrogen, helium, germanium, copper, charcoal, carbon, calcium, cadmium, bismuth, aluminium, *gold*
illness	smallpox, plague, meningitis, malnutrition, leukemia, hepatitis, glaucoma, flu, eczema, diabetes, cirrhosis, cholera, cancer, asthma, arthritis, anthrax, acne, *menopause*
feeling	wonder, shame, sadness, pleasure, passion, love, joy, happiness, fear, anger, *heaviness, coolness, torment, tenderness, suffering, stinging*
vehicle	van, truck, ship, rocket, pickup, motorcycle, helicopter, cruiser, car, boat, bicycle, automobile, airplane, aircraft, *jag*
creator	producer, photographer, painter, originator, musician, manufacturer, maker, inventor, farmer, developer, designer, craftsman, constructor, builder, artist, architect, *motivator*
pain	toothache, soreness, sting, soreness, sciatica, neuralgia, migraine, lumbago, headache, earache, burn, bellyache, backache, ache, *rheumatism, pain*
animal	zebra, turtle, tiger, sheep, rat, puppy, monkey, lion, kitten, horse, elephant, dog, deer, cow, cat, camel, bull, bear
game	whist, volleyball, tennis, softball, soccer, rugby, lotto, keno, handball, golf, football, curling, chess, bowling, basketball, baccarat, *twister*
edible fruit	watermelon, strawberry, pineapple, pear, peach, orange, olive, melon, mango, lemon, kiwi, grape, cherry, berry, banana, apple, *oyster, walnut, pistachio, mandarin, lime, fig, chestnut*

Figure 2: Optimal clustering for large evaluation set.

contexts used	purity
(VI)	73.4%
(VI), (II)	76.6%
(VI), (II), (III)	80.1%

Table 5: Clustering the larger evaluation set with an increasing number of context specifications.

given in Table 5. As can be seen the object relation is added first again. This time though the inclusion of adjectival modification brings another performance increase which is even one per cent above the result for the space built from all possible relations. The addition of any further contexts consistently degrades performance. The clustering solution thus produced is given in Figure 2. From the 1 872 698 dimension used in the original space only 341 214 are retained.

4 Discussion and Conclusion

Our results are counterintuitive at first sight as it could be expected that a larger number of different contexts would increase performance. Instead we see the best performance with only a very limited set of possible contexts. We suspect that this behaviour is due to a large amount of correlation between the different kinds of contexts. The addition of further contexts beyond a certain point therefore has no positive effect. As an indication for this it might be noticed that the three context specifications that yield the best result for the 402 word set comprise relations with the three main open word classes. It is to be expected that they contribute orthogonal information that covers central dimensions of meaning. The slight decrease in performance that can be observed when further contexts are added is probably due to chance fluctuations and almost certainly not significant; with significance being hard to determine for any of the results.

However, it is obviously necessary to cover a basic variety of features. Patterns which are used to explicitly track semantic relations on the textual surface seem to be too restrictive. Information accessible from co-occurring verbs for example is completely lost. In a regular window based word space such information is retained and its performance is competitive with a pattern based approach. This method is obviously too liberal, though, if compared to the dependency spaces.

In general we were able to show that semantic spaces are obviously able to capture categorical knowledge about concepts best when they are built from a syntactically annotated source. This is true even if the context specification used is not the most parsimonious. The problem of determining the right set of contexts is therefore rather an optimisation issue than a question of using dependency based spaces or not. It is a considerable one, though, as computations are much cheaper with vectors of reduced dimensionality, of course.

For the categorisation task the inclusion of more complex relations reaching over several dependencies does not seem to be helpful considering they can all be dropped without a decrease in performance. As Pado and Lapata (2007) reached better results in their experiments with a broader set of context specifications we conclude that the selection of the kinds of context to include when constructing a word space depends largely on the task at hand.

References

A. Almuhareb and M. Poesio. 2004. Attribute-based and value-based clustering: An evaluation. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 158–165, Barcelona, Spain, July. Association for Computational Linguistics.

M. Poesio and A. Almuhareb. 2005a. Concept learning and categorization from the web. In *Proceedings of CogSci2005 - XXVII Annual Conference of the Cognitive Science Society*, pages 103–108, Stresa, Italy.

A. Almuhareb and M. Poesio. 2005b. Finding attributes in the web using a parser. In *Proceedings of Corpus Linguistics*, Birmingham.

Marco Baroni, Stefan Evert, and Alessandro Lenci, editors. 2008. *ESSLLI Workshop on Distributional Lexical Semantics*, Hamburg, August.

J. R. Curran and M. Moens. 2002. Improvements in automatic thesaurus extraction. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*, pages 59–66, Morristown, NJ, USA. Association for Computational Linguistics.

T. Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

A. Ferraresi, E. Zanchetta, M. Baroni, and S. Bernardini. 2008. Introducing and evaluating ukwac, a very large web-derived corpus of english. In *Proceedings of the WAC4 Workshop at LREC 2008*.

G. Grefenstette. 1994. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, Dordrecht.

D. Hindle. 1990. Noun classification from predicate-argument structures. In *Meeting of the Association for Computational Linguistics*, pages 268–275.

G. Karypis. 2003. Cluto: A clustering toolkit. technical report 02-017. Technical report, University of Minnesota, November.

D. Lin. 1998. Automatic retrieval and clustering of similar words. In *COLING-ACL*, pages 768–774.

S. Padó and M. Lapata. 2007. Dependency-based construction of semantic space models. *Comput. Linguist.*, 33(2):161–199.

M. Poesio and A. Almuhareb. 2005b. Identifying concept attributes using a classifier. In *Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition*, pages 18–27, Ann Arbor, Michigan, June. Association for Computational Linguistics.

M. Sahlgren. 2006. *The Word Space Model*. Ph.D. thesis, Department of Linguistics, Stockholm University.

H. Schütze. 1992. Dimensions of meaning. In *Supercomputing '92: Proceedings of the 1992 ACM/IEEE conference on Supercomputing*, pages 787–796, Los Alamitos, CA, USA. IEEE Computer Society Press.